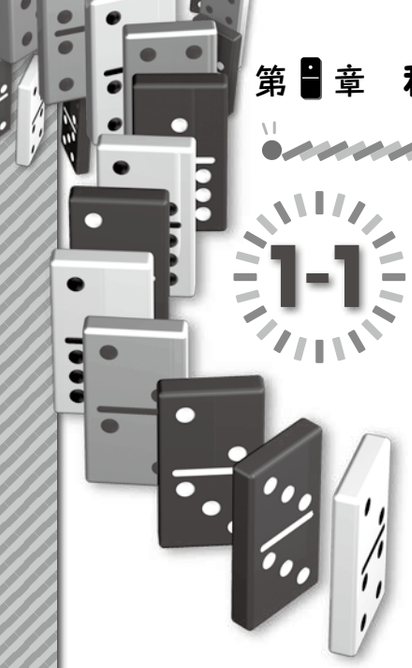


1-1 ビジネスの変革をもたらす機械学習

深層学習、人工知能との違いとは

本稿では、最近ビジネスに活用されている機械学習、深層学習、人工知能について概観します。それらがビジネスにどのように活用されているか、またその際に誤解されやすい事柄を整理します。まずは本稿を、機械学習をビジネスに応用するうえでの足がかりにしてもらいたいと思います。

Author 黒柳 敬一（くろやなぎ けいいち）Sansan㈱Data Strategy & Operation Center
Twitter @Keiku　Blog http://keiku.hatenablog.jp/
Web https://www.kaggle.com/keiku322

機械学習とは？

機械学習は、もともと人間の持つ学習能力を機械（計算機）に持たせることを目指す人工知能の一研究分野として発展してきました。機械学習とは一言で言うと、「機械（計算機）によってデータから何らかの法則（ルール）を学習（ルールを構築）し、その法則によって決められたパターンを得ること」です。

機械学習でできること

機械学習でできることは大きく分けて2つあります。それは、「予測」と「発見」です。冒頭で触れた得られるパターンというのは、これらを指します。この予測と発見は、それぞれ用いるデータの期間や目的などが異なります。

- 予測：過去から現在までのデータをもとに未来／未知のデータに対して予測する
- 発見：過去から現在までのデータから未知のパターンを発見する

予測は、機械学習の枠組みとしては「教師あり学習」と呼ばれます。学習において予測のお手本となる過去の正解データを用いて学習することから、このように呼ばれます。教師あり学習によっておもに、「分類」と「回帰」というタスクを実行できます。

分類というのは、過去のデータの分類傾向に基づいて未来のデータではどのように分類されるかを予測する方法です。一方で回帰というのは、過去のデータの連続値の推移から、未来のデータがどのような値になるかを予測する方法です。とくに時系列的に未来のデータに限らず、未知のデータに対して予測できます。

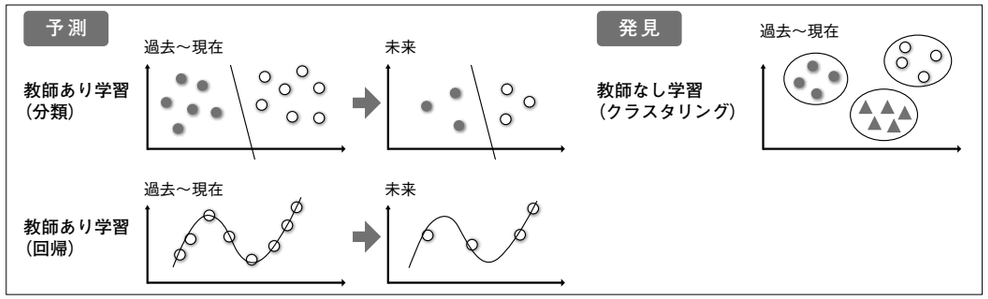
また発見は、機械学習の枠組みとしては「教師なし学習」と呼ばれます。教師あり学習とは異なり、学習において過去の正解データを用いることなく学習することから、このように呼ばれます。教師なし学習では、おもに「クラスタリング」というタスクを実行できます。クラスタリングというのは、似ているものをまとめることで新たな知見を得る方法です。

これらを概念図として表すと**図1**のとおりです。

今ビジネスで活用されている機械学習

機械学習の分類や回帰、またクラスタリングの手法は、さまざまな領域における課題に対して適用できます。そのためには、まずコンピュータが学習するための「データ」が必要です。データさえあればさまざまな課題に対応できるようになります。機械学習の適用事例を、領域ごとに**表1**に示します。これらの事例からもわかるように、金融、マーケティング、Webなどさまざまな領域で機械学習が活用されていることがわかります。

▼図1　機械学習でできること



さまざまな領域で活用されている背景としては、「ビッグデータ」の重要性が叫ばれる中、その蓄積と活用のためのインフラが整えられてきたことが大きな要因と言えます。また、「オープンソースソフトウェア（OSS）」の発展の寄与も大きく、機械学習を容易に使うことができるようになったことも理由の1つです。

機械学習による学習と予測

ここで機械学習における学習と予測についてイメージを持ってもらうために、やや正確性を犠牲にして直観的に解説します。

機械学習による学習は、「直線を引くイメージ」です。何か2つのクラスで分けられていることが知られているデータがあったとします。このデータは、たとえば、腫瘍（しゅよう）の悪性／良性、ECサイトでの購入あり／なし、迷惑メール／正常メールなどの区分です。このデータに含まれる特徴を軸にとり、**図2**のようにプロットしてみます。たとえば、腫瘍の例であれば、腫瘍のサイズ、人の性別、年齢などが特徴にあたります。このデータの2つのクラスをうまく分けられるような直線を引くことを考えます。

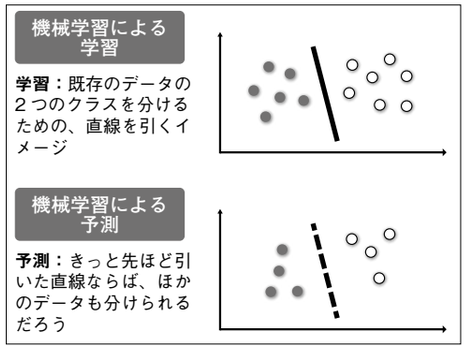
大雑把に言えば、この直線を自動で引くしくみが機械学習の「学習」です。具体的には、直線に一番近い黒丸／白丸双方への距離（マージンと言います）がなるべく大きくなる直線を引けるように学習します。この引かれた直線が別のデータでもきっと同じように分類できるだろうと思って予想して、別のデータにも直線を引くことが機械学習の「予測」です。これが機械学習における学習と予測のイメージです。

この学習と予測に利用できるアルゴリズムはさまざまあり、決定木、ロジスティック回帰、ランダムフォレスト、ニューラルネットワーク

▼表1　機械学習の事例

領域	機械学習の事例
金融	市場予測、与信審査における信用リスク評価、不正検知
マーケティング	需要予測、顧客セグメンテーション、ダイレクトメール（DM）のターゲティング、評判分析
Web	情報検索、スパムフィルタリング、レコメンデーション、機械翻訳、ソーシャルネットワーク分析
広告	広告のクリック率予測
ヘルスケア	MRI画像による医療診断
マルチメディア	音声認識、画像認識
機械・製造業など	故障・異常検知

▼図2　機械学習による学習と予測



など多数存在します。これらのアルゴリズムについてはどこかで耳にしたことがあるかもしれません。それぞれのアルゴリズムについての解説は、誌面の都合上、ほかの書籍に任せたいと思います。参考文献[1][2]を参照してください。

機械学習の性質を知る

表1で取り上げた機械学習の事例は、実はどれも人が判断できる法則ばかりです。ここで、人と比較することで機械学習にはどのような特徴があるのか説明します。

人と比較して機械学習の得意なところ

人と比較して、機械学習には次のような有利な点があります。

- 人手で処理できる量以上に「大量に処理」できる
- 人の処理速度以上に「高速に処理」できる
- 人が判断するよりも「高精度に判断」できることがある

しかも、コンピュータは疲れることを知りません。これらの利点が大きな要因となり、機械学習は実世界においても利用されています。

日常の生活でも、機械学習による大量・高速な処理の恩恵を実感できます。たとえば、迷惑メールを除外する機能であるスパムフィルタリングがあります。もしこの機能がなければ、途方もない数の迷惑メールを人手でひとつひとつ閲覧してゴミ箱に捨てなければなりません。しかし機械学習を用いれば、容易に大量・高速に迷惑メールを除外できます。

また、機械学習が高精度に判断できる最近の事例として、Facebook社によるDeepFaceという人間の顔認証技術が、人と同等以上の顔認証の精度を実現しています。特定の分野については、人の判断や処理に要する時間やその精度を上回っていることは明らかです。とくに「深層学習」と呼ばれる機械学習の技術が、高精度に判断できる要因となっています。このような背景もあり現在、深層学習の研究開発が盛んに行われています。

人と比較して機械学習の苦手なところ

逆に、人と比較して機械学習を用いるには難しい場面もあります。

- 「少ない情報から推論する」ことが得意ではない
- 状況の変化に対して柔軟に対応する

データが少ないと機械学習を利用することは難しいでしょう。人は少ない情報から推論することが得意ですが、これは機械学習では苦手な処理です。冒頭でも触れましたが、やはりデータが大量にあることで機械学習は力を発揮します。

また機械学習のロジックは、アルゴリズムの組み合わせでしかないので、ルールに当てはまらないような処理に対応することが苦手です。たとえば、再びスパムフィルタリングの例を考えてみましょう。スパムフィルタリングも万能ではなく、度々、人が正常であると判断されるメールについても迷惑メールフォルダに振り分けられていることがあると思います。この問題が生じる理由としては、過去のデータから学習したルールに当てはまらない事象が生じてしまったことが原因となっています。

深層学習とは？

以下、やや正確性に欠けるかもしれませんが、深層学習を簡単に理解してもらうために、機械学習における深層学習の位置づけと、簡単な歴史を解説します。より詳しい解説は参考文献[3][4]を参照してください。

深層学習の位置づけ

深層学習は、そもそも機械学習における一分野であるニューラルネットワークが発展して形成された分野であり、**図3**のような枠組みとして概観できます。そのため、「機械学習、その中でもニューラルネットワークを理解したあとに深層学習は取り組むべきものであって、機械学習入門者がいきなり取り組む分野ではない」ことを注意しておきます。

深層学習小史――深層（ディープ）までの道のり

ニューラルネットワークの歴史は古く、1940年代には研究が開始されていた分野であり、最近突如登場した技術ではありません。

ニューラルネットワークは、脳の一部（ニューロン）の構造を単純化して数学的にモデル化した「形式ニューロン」[注1]から構成されるネットワークです。ニューラルネットワーク研究のおもなブレイクスルーとしては、次が挙げられます（**図4**）。

- 1958年のFrank Rosenblattによるパーセプトロン
- 1986年のDavid E. Rumelhartによるバックプロパゲーションネットワーク
- 2006年のGeoffrey Everest Hintonらによるディープビリーフネットワークなど

これらの発表に伴い、ブームが起きては下火になるということを繰り返しています。とくに2006年ごろからこれまで実現し得なかった多層のニューラルネットワークが実現されたことで、画像認識、音声認識、機械翻訳などさまざまな領域のタスクで、次々と劇的な精度向上が見られるようになります。これを機に「深層学習（Deep Learning）」という分野として確立されます。そして、現在も活発に研究がなされています。

注1） いくつかの入力に対して入力の合計が一定以上になると発火（出力）が起きるしくみ。

▼図3　深層学習の位置づけ

機械学習と深層学習の混同されやすいところ

昨今の深層学習ブームによって深層学習という言葉が独り歩きして、クローズアップされがちです。そこで従来の機械学習技術と比較して、深層学習に対して勘違いされやすい事柄を整理します。

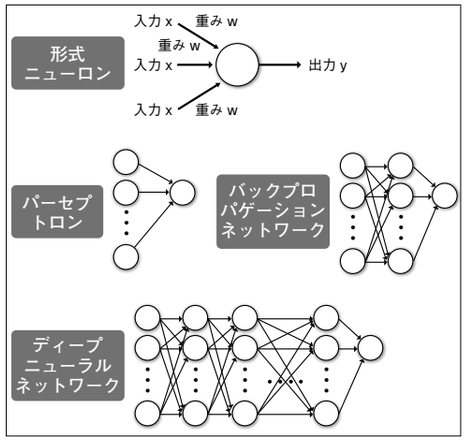
深層学習なら自動で特徴抽出を行える？

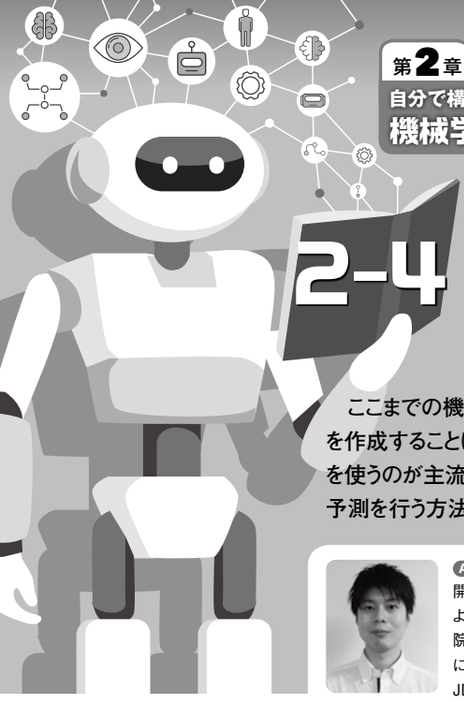
画像認識における深層学習の技術として、畳み込みニューラルネットワークという技術があります。この技術により、従来の画像認識技術（局所特徴抽出やコーディング手法など）を要することなく、そのままの画像から学習することが可能となったことで、自動で特徴抽出できるという解説をよく目にするかと思います。

しかしこれが意味するところは、画像データに対する畳み込みニューラルネットワーク技術の特徴であって、あらゆるデータ（社内の文章や、売上などの数値データなど）に対して自動で特徴抽出を行えるということは意味していません。

また、対象が画像であれば何でも自動で都合の良い特徴抽出をしてくれる技術ではありませ

▼図4　ニューラルネットワークの発展





2-4 本格的にPythonライブラリを使おう

scikit-learn入門と機械学習の勉強方法

ここまでの機械学習クラウドサービスではもの足りない場合には、自分で学習モデルを作成することになります。現在は、機械学習関連のライブラリが充実しているPythonを使うのが主流です。本稿では、scikit-learnというライブラリを使ってモデルを作成し予測を行う方法と、そのために必要な知識・勉強法について解説します。

Author 小川 雄太郎（おがわ ゆうたろう） 株式会社電通国際情報サービス（ISID）、技術本部開発技術部に所属。ディープラーニングをはじめとした機械学習関連技術の研究開発・技術支援、およびHR techに関するデータ解析を業務とする。明石高専、東京大学工学部を経て、東京大学大学院新領域創成科学研究科、神保・小谷研究室にて、脳機能計測および計算論的神経科学の研究に従事し、2016年博士号（科学）を取得。東京大学特任研究員を経て、2017年4月より現職。JDLA DeepLearning for GENERAL 2017。 Mail ogawa.yutaro@isid.co.jp

はじめに

本稿では機械学習を自分で実装できるようになるために必要な知識、環境構築方法、アルゴリズムの解説と実装例、そして機械学習の勉強方法を紹介します。

機械学習・AIを使用している企業はおもに2種類あり、㈱Gunosyやクックパッド㈱のように自社サービスに適用している企業と、㈱電通国際情報サービス（ISID）のようにお客様の機械学習・AI活用システムをSIerとして構築する企業があります。本稿の内容は、弊社ISIDの機械学習チームが社内教育で提供および提供予定の資料から紹介します。

機械学習ライブラリ scikit-learn

Pythonとscikit-learnが使われる理由

機械学習を実装する際には、言語としてPython、ライブラリとしてscikit-learn（サイキット・ラーン）を使用することが一般的です。もちろんRやJavaでも実装可能ですが、本稿では最もポピュラーなPythonとscikit-learnを用いた機械学習の実装を紹介します。

Pythonおよびscikit-learnが機械学習に頻繁に使用されているのには2つの理由がある、と個人的に感じています。

1つめの理由は、Pythonがオブジェクト指向言語の中でも比較的シンプルであることです。そのためITエンジニアに比べてプログラミングスキルが低いアカデミック領域の研究者でも、機械学習を研究したり使用したりする際に実装が容易という利点がありました。

2つめの理由は数値計算や数値解析のライブラリが整備されていたことです。Pythonには行列の掛け算を扱う命令はありません。しかし、NumPy（ナンパイ）というライブラリが、行列演算をはじめ数値計算に必要なクラスやメソッドを提供してくれます。また、NumPyはC言語およびFORTRANをベースに書かれており、実行が高速という利点もあります。そして、このNumPyをベースとして、行列の固有値を求めたり、微分方程式を解いたりするなどの数値解析が実行できるSciPy（サイパイ）というライブラリも提供されています。

これら2つの理由からアカデミック領域の研究者が慣れている商用技術計算用言語MATLABとほぼ同機能の実装環境がPython、NumPy、SciPyをベースにして無料の環境で実現できました。さらに、Google Summer of Codeというイベントや機械学習研究者の協力を通じて、最終的に機械学習ライブラリscikit-learnが作られました。

scikit-learnを使うために必要なライブラリ

scikit-learnはPythonで書かれているため、Pythonが実行できる環境が必要です。環境構築の方法についてはのちほど紹介します。また、NumPy、SciPyライブラリに依存しているため、これら2つも必須となります。

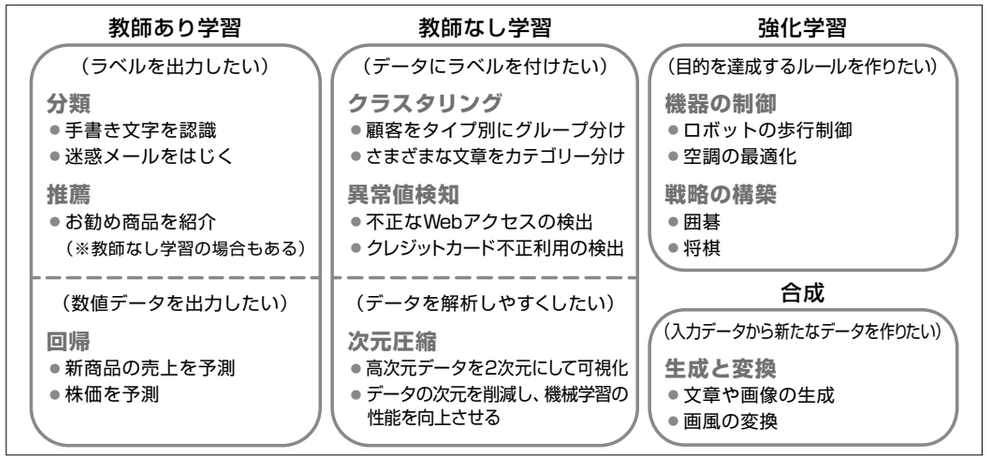
そのほかにscikit-learnを使う際に一緒に使用される代表的なライブラリとして、Matplotlibとpandasがあります。Matplotlibはグラフを描画するためのライブラリです。商用技術計算用言語MATLABでのグラフ描画命令と同様の形式で、Pythonでグラフを描画できるためMatplotlibと呼ばれます。pandasはCSVデータなどをPythonで扱いやすくするためのライブラリです。ExcelやCSVなどの表形式データはパネルデータと呼ばれ、その頭文字をとってpan（el）-da（ta）-s→pandasと呼ばれています。動物のパンダとは関係ないようです。

scikit-learnで扱えるアルゴリズム

2-1でも紹介されたように機械学習は教師あり学習、教師なし学習、強化学習と3分類されます。最近は生成アルゴリズムの研究も多いので（その実は合成であり、ほかの3つの機械学習手法がベースですが……）、筆者の頭の中では**図1**のように4分類で整理しています。**図1**は機械学習の大分類と小分類、そして活用例を掲載しています。言葉の定義ですが、データとは画像（縦ピクセル数×横ピクセル数のRGB値）や、文章のことを指します。ラベルとはデータに対して人が定めた識別子のことを指します。たとえば、手書き文字の1は画像データであり、それを人が1であるとラベル付けしています。推薦はアルゴリズムによっては教師なし学習に分類することも多いですが、個人的には教師あり学習にくくっています。機械学習の分類は個人ごとに思想が違うので正解がないのですが、1つの参考にしていただければと思います。

機械学習の中でもscikit-learnが得意なのが、教師あり学習の分類と回帰、教師なし学習のクラスタリングと異常値検知、そして次元圧縮です。一方で、推薦や強化学習、合成は別ライブラリのほうが優れています（できないことはないですが）。日本語の自然言語処理の場合、MeCab、JUMAN、CaboCha、gensimなどの別ライブラリと組み合わせて解析することが一般的です。また、強化学習や合成は機械学習の中

▼図1 機械学習の分類の一例



でも毛色が異なります。強化学習については、筆者が「作りながら学ぶ強化学習」[注1]というWeb連載で詳しく解説していますので、そちらをご覧ください。

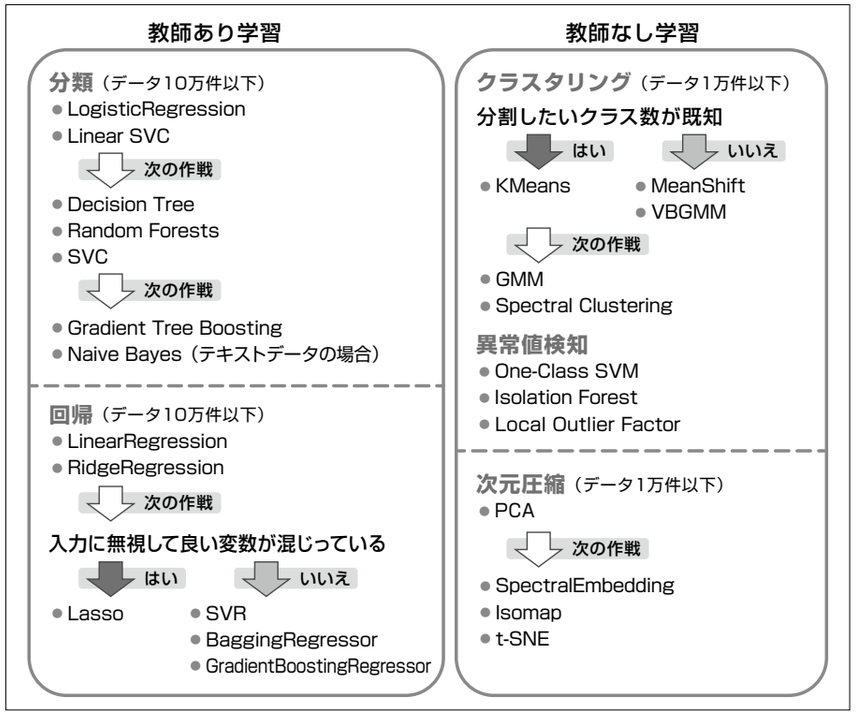
scikit-learnを使用するにあたり、どのアルゴリズムを使用すれば良いのかの選定は、scikit-learnのChoosing the right estimatorページ[注2]を見ると整理されています。ただし、このページは英語表記であり、多少扱いづらい点もあるので、筆者なりに整理したのが**図2**です。この図では分類や回帰など目的別に大項目が分かれています。目的名の横に記載したデータ件数は、目安となるデータ数の上限です。上限を超える大規模データの場合は、この図には記載していないアルゴリズムを検討する必要があります。各大項目の下には、まず使用すべき基本的アルゴリズムを記載しています。矢印の「次の作戦」で示しているアルゴリズムは、基本的アルゴリズムでうまく機能しない場合や、より高い性能を求める際に試みます。アルゴリズムの分類・整理も個人ごとに思想が違うので正解はありませんが、1つの参考にしていただければと思います。なお、近年大注目のDeep Learningは機械学習のアルゴリズムの1つであるNeural Networkのさらに一部という位置づけになります。

本稿ではこれらのアルゴリズムの中から、教師あり学習の分類アルゴリズムである「ロジスティック回帰（LogisticRegression）」について実装例とアルゴリズムの解説を紹介します。

ロジスティック回帰を選んだ理由は、このアルゴリズムがKaggle[注3]と呼ばれるデータサイエンティストのコンペティションやビジネスの現場で頻繁に使用されている、重要かつ基礎的なアルゴリズムだからです。Kaggleとは企業などがデータセットと課題を設定し、その解決アルゴリズムをユーザが提出して性能を競うコンテストサイトです。最近では、㈱メルカリが出品商品の紹介テキストデータから適正価格を予想するアルゴリズムを募集し、優勝チームに6万ドルを支払うコンペを開催したことでも有名になりました。

注1） URL https://book.mynavi.jp/manatee/series/detail/id=87626
注2） URL http://scikit-learn.org/stable/tutorial/machine_learning_map/
注3） URL https://www.kaggle.com/surveys/2017

▼図2　scikit-learnのアルゴリズムチートシート（小川版）



機械学習を実装する環境を整える

Pythonおよびscikit-learnで機械学習を実装するための環境構築方法を紹介します。実装環境を整えるには、Webブラウザ上で実装できる無料サービスを利用する方法と、自分のPCのローカル環境に実装環境を整える方法の2パターンがあります。最初に体験してみるだけであれば、Webブラウザ上で実装できる無料サービスを使用するのをお勧めします。その後、より複雑で大規模なことをやりたくなったらローカル環境を構築するのが良いと思います。**表1**に機械学習の実装・実行方法をまとめました。順番に説明します。

▼表1　自前で機械学習を実装・実行する方法

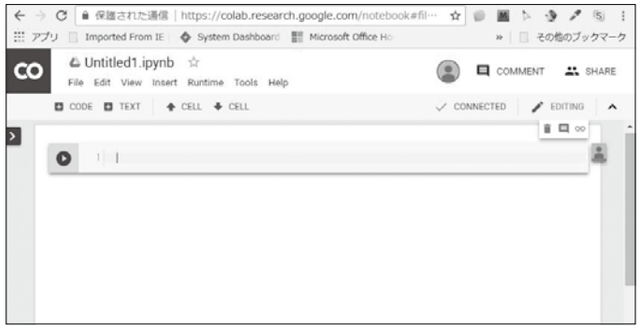
実装・実行環境	ツール名	特徴
Webブラウザ上	try Jupyter！	無料ですぐに使用可能
	Google Colaboratory	無料だがGoogleアカウントが必要
ローカル環境	Jupyter Notebook	対話型実行環境
	PyCharmなど	IDE（統合開発環境）
	Atomなど	エディタ

Webブラウザ上で実装・実行する環境

Webブラウザ上でPythonおよびscikit-learnを使用して機械学習を実装できるサイトとして有名なのが、try Jupyter！[注4]とGoogle Colaboratory[注5]です。シンプルな実装を体験したいだけならすぐに使用できるtry Jupyter！がお勧めです。一方で、Google Colaboratory（**図3**）はGoogleアカウントの用意が必要で、推奨ブラウザはChrome、Firefoxです。以前は申し込みから使用できるようになるまでに1〜2日程度かかりましたが、現在はすぐに使えるようです。GPUが使用できて実行が速い、pipと呼ばれるライブラリ追加・管理のコマンドが使用できて好きなPythonライブラリを追加できる、Google Driveからデータを持ってきやすい、コードを他人とシェアしやすい、などのメリットがあります。本稿ではGoogle Colaboratoryを使用して、機械学習アルゴリズムの実装を紹介します。

ローカル環境でPythonおよびscikit-learnで機械学習を実装する手順は誌面の都合上割愛します。本記事の最後で紹介する書籍で詳しく紹介されているのでそちらをご覧ください。

▼図3　Google Colaboratoryの画面



機械学習を実装・実行するツール

Pythonおよびscikit-learnなどのライブラリを使用して機械学習を実装し、実行する方法は大きく2つあります。実装・実行を対話型で行えるツールを使用する方法とエディタを使用してコーディングし、コマンドプロンプトからPythonファイルを実行する方法です。

注4） URL https://try.jupyter.org/
注5） URL https://colab.research.google.com/